1 Granular-ball Generation Process

This part presents the process and details of granular-ball update. First, Figure 1 provided a schematic illustration of granular-ball splitting. Figure 1 illustrates that, for a granular-ball GB_i, if its CCM_{gb} value exceeds twice the $\text{CCM}_{\text{median}}$ value of all granular-balls in the current splitting stage, the splitting process is terminated. Otherwise, it is further divided into two sub-balls and proceeds to the next splitting stage. The reason for using the median as the threshold criterion is that it provides a robust estimate of the central tendency, which is less sensitive to outliers and skewed distributions, ensuring a more stable and reliable splitting decision. Secondly, Algorithm 1 provides the pseudo-code for granular-ball update.

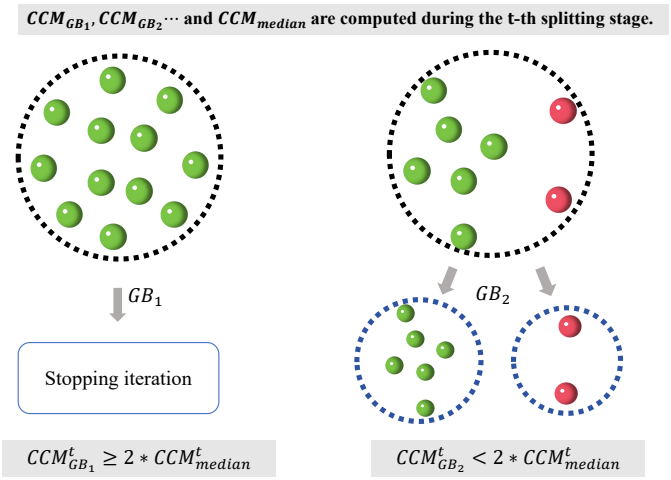

Figure 1: Examples of CCM_{gb}.

Algorithm 1 Granular Ball Generation Process

Input: Dataset $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$.
Output:The final GB.

```
Treat the entire dataset as a whole.
Coarsely divide the data into initial granular balls GB_i(i = 1,2,...,√n) by k-means.
while the granular ball set is not stable do
    for each GB_i ∈ {GB_1, GB_2, ..., GB_√n} do
        Compute CCM_{GB_i} using Equation 3 and its median value CCM_median.
        if CCM_{GB_i} < 2 · CCM_median then
            Call the 2-means algorithm to split GB_i into two sub-balls GB_i1 and GB_i2.
            if |GB_i1| > 1 and |GB_i2| > 1 then
                Add GB_i1 and GB_i2 to the granular ball set.
            else
                Retain the original GB_i.
            end if
        end if
    end for
end while
Get the final set of granualr ball.
return GB.
```

2 Proof of Differentiability and Convexity

The analysis of the differentiable and convexity of the objective function ensures that $\Gamma(\sigma) = \max_{\mathbf{H}_{gb}\in\Omega} \text{Tr}\left[\mathbf{K}_\sigma^{GB}(\mathbf{H}_{gb}\mathbf{H}_{gb}^T - \mathbf{I}_m)\right]$ can be optimized through numerical optimization methods, leading to a stable and globally optimal solution. Below, specific proofs are provided in Theorems 1 and 2.

Theorem 1. *The objective function $\Gamma(\sigma)$ is differentiable.*

Proof. First, since $\mathbf{K}_{\boldsymbol{\sigma}}^{GB} = \sum_{k=1}^{\omega} \sigma_k^2 \mathbf{K}_k^{GB}$ is a linear combination of σ_k, we have for every σ_k:

$$\frac{\partial \mathbf{K}_\sigma^{GB}}{\partial \sigma_k} = 2\sigma_k \mathbf{K}_k^{GB}, \tag{1}$$

which shows that $\mathbf{K}_\sigma^{GB}$ is differentiable with respect to σ_k.

Second, since $\Gamma(\sigma)$ depends on $\mathbf{K}_\sigma^{GB}$, using the chain rule, we have:

$$\begin{aligned}\frac{\partial \Gamma(\sigma)}{\partial \sigma_k} &= \frac{\partial}{\partial \sigma_k} \max_{\mathbf{H}_{gb}\in\Omega} \text{Tr}[\mathbf{K}_\sigma^{GB}(\mathbf{H}_{gb}\mathbf{H}_{gb}^T - \mathbf{I}_m)] \\ &= \max_{\mathbf{H}_{gb}\in\Omega} \frac{\partial}{\partial \sigma_k} \text{Tr}[\mathbf{K}_\sigma^{GB}(\mathbf{H}_{gb}\mathbf{H}_{gb}^T - \mathbf{I}_m)],\end{aligned}$$

and substituting $\mathbf{K}_{\boldsymbol{\sigma}}^{GB} = \sum_{k=1}^{\omega} \sigma_k^2 \mathbf{K}_k^{GB}$, we get:

$$\frac{\partial \Gamma(\sigma)}{\partial \sigma_k} = 2\sigma_k \max_{\mathbf{H}_{gb}\in\Omega} \text{Tr}[\mathbf{K}_k^{GB}(\mathbf{H}_{gb}\mathbf{H}_{gb}^T - \mathbf{I}_m)]. \tag{2}$$

Therefore, $\Gamma(\sigma)$ is also differentiable with respect to σ_k, and the form of its partial derivative depends only on the granular-ball kernel matrix $\mathbf{K}_\sigma^{GB}$ and the granular-ball distribution matrix $\mathbf{H}_{gb}$. Theorem 2 indicates that the objective function has sufficient smoothness to guide the update direction and step size of parameter optimization. □

In summary, the objective function $\Gamma(\sigma)$ based on the granular-ball kernel matrix exhibits excellent smoothness and convexity. The multi-granularity representation of granular balls and the optimization of the parameter σ_k provide a more flexible capability for multi-kernel data modeling.

Theorem 2. *The objective function $\Gamma(\sigma)$ is convex.*

Proof. For $\sigma_1, \sigma_2 \in \Delta$ and $0 < \alpha < 1$, we have:

Proof: For any $\sigma_1, \sigma_2 \in \Delta$, and $0 < \alpha < 1$, we have:

$$\begin{aligned}&\Gamma[\alpha\sigma_1 + (1-\alpha)\sigma_2] \\ &= \alpha\Gamma(\sigma_1) + (1-\alpha)\Gamma(\sigma_2) \\ &= \alpha\sigma_1 + (1-\alpha)\sigma_2 \max_{\mathbf{H}_{gb}\in\Omega} \text{Tr}[\mathbf{K}_\sigma^{GB}(\mathbf{H}_{gb}\mathbf{H}_{gb}^T - \mathbf{I}_m)] \\ &= \max_{\mathbf{H}_{gb}\in\Omega} \text{Tr}\left[\sum_{k=1}^{\omega} \left(\alpha\sigma_{1,k}^2 + (1-\alpha)^2\sigma_{2,k}^2\right) \mathbf{K}_k^{GB}(\mathbf{H}_{gb}\mathbf{H}_{gb}^T - \mathbf{I}_m)\right].\end{aligned}$$

Since $\sigma_{1,k}, \sigma_{2,k} \geq 0$, based on the properties of linear combinations and matrix inequalities in convex optimization, we can obtain:

$$
\begin{aligned}
&\max_{\mathbf{H}_{gb}\in\Omega} \operatorname{Tr}\left\{\sum_{k=1}^{\omega}\left[\alpha^2\sigma_{1,k}^2+2\alpha(1-\alpha)\sigma_{1,k}\sigma_{2,k}+(1-\alpha)^2\sigma_{2,k}^2\right]\right. \\
&\qquad\left.\cdot\ \mathbf{K}_k^{GB}(\mathbf{H}_{gb}\mathbf{H}_{gb}^T-\mathbf{I}_m)\right\} \\
&\le \max_{\mathbf{H}_{gb}\in\Omega} \operatorname{Tr}\left\{\sum_{k=1}^{\omega}\left[\alpha^2\sigma_{1,k}^2+(1-\alpha)^2\sigma_{2,k}^2\right]\right. \\
&\qquad\left.\cdot\ \mathbf{K}_k^{GB}(\mathbf{H}_{gb}\mathbf{H}_{gb}^T-\mathbf{I}_m)\right\} \\
&\le \alpha \max_{\mathbf{H}_{gb}\in\Omega} \operatorname{Tr}\left\{\sum_{k=1}^{\omega}\sigma_{1,k}^2\mathbf{K}_k^{GB}(\mathbf{H}_{gb}\mathbf{H}_{gb}^T-\mathbf{I}_m)\right\} \\
&\quad+(1-\alpha)\max_{\mathbf{H}_{gb}\in\Omega} \operatorname{Tr}\left\{\sum_{k=1}^{\omega}\sigma_{2,k}^2\mathbf{K}_k^{GB}(\mathbf{H}_{gb}\mathbf{H}_{gb}^T-\mathbf{I}_m)\right\}.
\end{aligned}
\tag{3}
$$

Simplifying, we derive:

$$
\Gamma[\alpha\sigma_1+(1-\alpha)\sigma_2]\le\alpha\Gamma(\sigma_1)+(1-\alpha)\Gamma(\sigma_2). \tag{4}
$$

Thus, $\Gamma(\sigma)$ is convex. Theorem 3 demonstrates that the objective function has a globally optimal solution. □

3 Supplementary of Full Results.

4 Friedman Test

We employ the well-known Friedman test to compare and analyze the significant differences among 12 algorithms across 9 datasets. The average ranking and accuracy of these algorithms on the 9 datasets are presented in Table 1.

Table 1: Average accuracy and ranks of 12 algorithms on 9 datasets

Method	Avg.ACC	Avg.rank
Random Selection	58.84	9.44
Kmeans	64.73	7.56
BKHK	67.21	7.11
RKKM-a	62.70	9.33
AASC	49.48	10.33
RMKKM	71.98	5.56
MKKM	72.79	4.78
GB-MKKM	73.84	5.22
MKKM-SR	66.54	7.44
GB-MKKM-SR	71.93	5.56
SMKKM	73.71	3.89
GB-SMKKM	80.54	1.78

Next, the formula for Friedman statistical variables is as follows:

$$
\chi_F^2=\frac{12n}{H(H+1)}[\sum_i^H R_i^2-3n(H+1)]=48.80. \tag{5}
$$

where H is the number of algorithms and n is the number of the datasets. R_i represents the average ranking of the i algorithm on the 9 data sets. In addition, according to the χ_F^2 distribution with $H-1$ degrees of freedom.

$$
F_F=\frac{(n-1)\chi_F^2}{n(H-1)-\chi_F^2}=7.78. \tag{6}
$$

where $F_F((H-1),(H-1)(n-1))$ obeys the F-distribution, and its degree of freedom is $(H-1)$ and $(H-1)(n-1)$. In this paper, we choose $\alpha=0.05$ and we can get $F_\alpha(11,88)=2.07$. Obviously, F_F is larger than F_α, So we can reject the null hypothesis at a confidence level of 95%, indicating that the 12 algorithms proposed in this paper are significantly different and not random.

Next, we use the Nemenyi post-hoc test to compare the errors of the 12 algorithms. If the average rank difference between two algorithms exceeds the critical value, their performance is considered significantly different. A larger difference indicates a more pronounced performance gap. The calculated threshold is $q_\alpha=3.268$, and the critical difference (CD) is computed as follows:

$$
CD=q_{\alpha=0.05}\sqrt{\frac{H(H+1)}{6n}}=3.268\times\sqrt{\frac{12(12+1)}{12\times 9}}=5.56. \tag{7}
$$

5 Convergence Analysis

Convergence serves as a fundamental criterion for validating the theoretical soundness of the proposed algorithm. The convergence behaviors on the four datasets are illustrated in Figure 2.

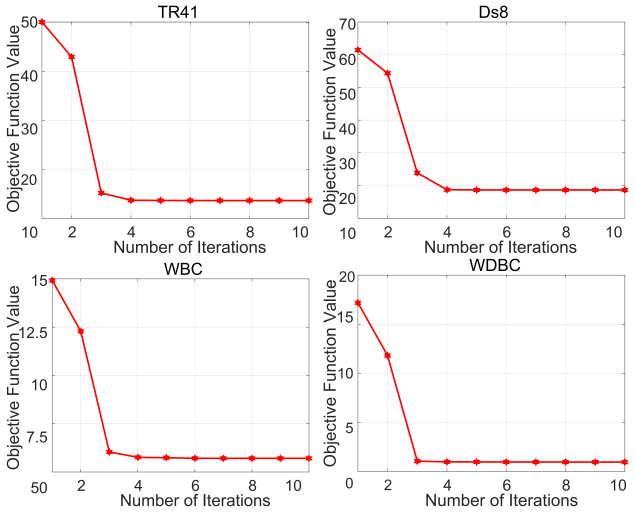

Figure 2: The iteration-wise changes in the objective values of GB-SMKKM on datasets.

As illustrated in Figure 2, the objective function exhibits a monotonically decreasing trend during the iterative process, which empirically confirms the convergence of the proposed algorithm.

6 Other Experimental Results

6.1 The Running Time Include Granular-ball Constuction

Granular-ball construction is performed as a pre-processing step that is independent of the subsequent model optimization process. Therefore, we do not include its runtime in the main performance comparison. As shown in Table1, both the ball generation and the overall optimization processes remain efficient, and the total runtime improves as the dataset size increases. This demonstrates the scalability and practicality of our method for large-scale applications.

Table 2: Running time (s).

Dataset	N	dims	SMKKM	Ball-generation	GB-SMKKM
COIL20	1440	1024	8.06	4.12	3.84
segment	2310	18	17.46	4.27	12.38
misplice-2	3175	2	34.12	8.98	6.52
Phoneme	5404	5	130.29	17.20	28.30 (much faster)

6.2 Experiments on KKM

Although our method is primarily designed to enhance and extend the Multiple Kernel k-means (MKKM) framework, we also include Kernel k-means (KKM) in our experiments for completeness and deeper insight. As a classical single-kernel clustering method, KKM provides a strong baseline for evaluating the benefits of multiple kernel integration. Including KKM serves two purposes: (1) it allows us to demonstrate that our improvements are not only effective in the multi-kernel setting, but also outperform strong single-kernel methods; and (2) it helps isolate the specific advantages brought by the granular-ball construction and adaptive integration mechanisms, beyond the general benefits of kernelization. Therefore, supplementing the experimental results with KKM adds value by highlighting the necessity of moving from traditional kernel clustering toward more flexible and robust multi-kernel solutions like ours. Table 3 reports the clustering performance of KKM.

Table 3: GB-KKM and KKM with the linear kernel (regarding ACC).

Dataset	N	dims	KKM	GB-KKM
movement-libras	360	90	46.58±0.027	54.23±0.034
Yeast	1484	7	32.91±0.003	43.47±0.004
misplice-2	3175	2	71.62±0.001	71.91±0.008
ls	6435	36	65.01±0.019	66.47±0.019

Table 4: Comparison results in terms of ACC on nine datasets.

Method	YALE	JAFFE	ORL	TR41	Srbct	GLIOMA	WBC	WDBC	Ds8	Average
Random Selection	50.30	58.69	65.50	48.86	47.62	46.00	81.72	62.57	68.36	58.84
Kmeans	43.03	80.28	65.00	55.47	52.38	64.00	81.55	87.70	53.13	64.73
BKHK	51.52	53.05	67.25	44.53	68.25	64.00	95.02	83.66	77.58	67.21
RKKM-a	41.06	62.77	48.15	47.84	49.84	50.90	80.40	85.64	92.29	62.10
AASC	40.64	30.99	27.09	19.68	34.76	53.90	76.57	62.92	98.75	49.48
RMKKM	52.18	87.07	55.58	62.85	53.89	59.30	96.49	86.95	93.53	71.98
MKKM	51.55	93.90	**68.36**	56.51	68.65	59.70	95.51	88.05	72.88	72.79
GB-MKKM	53.79	95.11	62.53	56.54	78.64	65.00	95.21	83.78	73.94	73.84
MKKM-SR	47.27	79.34	59.75	60.48	68.25	56.00	60.32	86.99	80.47	66.54
GB-MKKM-SR	47.69	96.55	65.85	52.50	68.52	52.63	95.78	87.10	80.76	71.93
SMKKM	**54.30**	95.87	66.69	56.65	66.27	59.40	96.93	88.40	78.87	73.71
GB-SMKKM	53.46	**97.00**	65.34	**64.63**	**82.14**	**74.47**	**97.56**	**91.07**	**99.18**	**80.54**

Table 5: Comparison results in terms of NMI on nine datasets.

Method	YALE	JAFFE	ORL	TR41	Srbct	GLIOMA	WBC	WDBC	Ds8	Average
Random Selection	52.00	73.26	79.27	41.14	36.50	29.24	39.68	19.51	35.49	45.12
Kmeans	51.63	69.05	81.49	38.32	49.05	40.84	69.70	37.92	35.84	52.65
BKHK	43.69	82.38	77.57	53.39	41.74	49.37	72.91	47.55	57.01	58.40
RKKM-a	46.01	70.17	68.44	42.91	29.73	28.22	41.09	44.93	82.27	50.42
AASC	46.83	27.08	43.65	5.88	5.30	43.36	15.91	0.26	92.41	31.19
RMKKM	55.58	89.37	74.84	63.52	31.35	49.82	76.51	46.05	76.16	62.58
MKKM	52.98	93.96	82.44	57.94	50.65	40.65	82.13	47.09	58.37	62.91
GB-MKKM	**65.76**	94.82	82.20	60.99	69.10	53.55	73.29	35.38	62.37	66.38
MKKM-SR	51.41	85.55	77.48	61.60	55.43	36.70	2.13	41.63	52.26	51.58
GB-MKKM-SR	62.08	95.76	82.92	59.16	56.45	44.08	74.76	41.16	53.94	63.37
SMKKM	56.11	95.18	82.06	57.57	47.85	41.20	79.71	45.86	57.71	62.58
GB-SMKKM	65.09	**97.80**	**83.89**	**64.07**	**82.31**	**61.22**	**85.75**	**53.67**	**94.76**	**76.51**

Table 6: Comparison results in terms of ARI on nine datasets.

Method	YALE	JAFFE	ORL	TR41	Srbct	GLIOMA	WBC	WDBC	Ds8	Average
Random Selection	27.58	49.94	49.18	23.34	23.46	14.88	40.14	4.77	26.53	28.87
Kmeans	27.55	47.07	53.31	14.96	31.26	27.97	80.83	44.05	40.06	40.78
BKHK	20.38	72.92	48.94	36.85	22.82	36.04	83.55	56.11	55.33	48.10
RKKM-a	43.58	66.83	52.85	63.95	56.19	53.40	80.40	85.64	96.64	66.61
AASC	42.33	32.51	31.49	30.40	43.97	57.00	76.57	62.92	98.75	52.88
RMKKM	53.64	88.90	60.20	77.61	56.67	64.30	96.47	86.95	93.53	75.36
MKKM	52.79	93.99	71.76	74.83	70.79	62.00	97.51	88.05	92.97	78.30
GB-MKKM	56.44	95.29	69.26	77.06	78.86	67.50	95.21	83.78	93.25	79.63
MKKM-SR	50.30	84.04	64.00	76.54	69.84	58.00	65.01	86.99	87.42	71.35
GB-MKKM-SR	49.23	95.76	70.73	71.94	78.95	57.89	95.78	87.10	87.37	77.19
SMKKM	55.00	95.87	70.55	74.46	68.25	63.30	96.93	88.40	88.93	77.97
GB-SMKKM	**56.92**	**97.95**	**72.00**	**78.61**	**87.38**	**74.47**	**97.56**	**91.07**	**99.18**	**83.90**